Figure 18 blocks with the same functions as corresponding blocks in Figures 15 and 17 have the same reference designations. The circuit differs from that shown in Figures 15 and 17 in that there is a comparator 90 which is used to detect the extension code. If there is a match, the comparator triggers a subtractor 100 which counts down the first counter 20 by 3 counts. When this has been done the extended length field, or specifically the data therein, is again written into the register 30. The various sizes associated with the extended field 14 must be added to the RAM memory 70. This implies that the number of cell sizes in the RAM memory will be doubled. In practice this means that a new memory bank will be used in the RAM memory 70. Unit 110 is a D-latch which latches the output value of the comparator 90 and uses it to address the new memory bank, in the RAM memory 70.

The comparator 90 and the subtractor 100 are the units that will handle the extended length field 14 so that the position in the header will be moved when the extension code is detected. Three extra bits will be added to the length field 11 and it is these extra bits that will be used to indicate the cell length. Accordingly the fixed size length field 11 is replaced with the extended length field 14 which is inserted into the data stream.

Compared with the operation of the circuit in Figure 15 or 17 where a field is written into the memory, in Figure 18 another field is written into the memory 70.

The cell header reading device shown in Figure 18 can also be used in order to implement the extension bit method. This is indicated in Figure 19. From the register 30 that contains the fixed size length field 11 the extension bit 13 is extracted and is used to increase the address range. The extension bit will count down the first counter 20 with three bits, indicated by the subtractor 100. This implies that three new bits will be written into register 30 and these new three bits plus the old three bits, i.e. altogether six bits, are

used to address the RAM memory 70 as symbolized by the six arrows. In this manner the number of cell sizes has been increased.

The ROM memory 40 may have several different mapping tables of the kind shown in Figure 5. It is possible to change from one mapping table to another in response to a predefined length code provided in the header of a mini cell. In this mammer it will be possible to switch from a first set of mini cell lengths, for example 4, 8, 16, 20 to a second set of lengths, for example 3, 6, 9, 12. Instead of using a ROM memory 40 configured with the mapping table shown in Figure 5 a RAM memory can be used for the same purpose. This will enable the control system 80 to write in new a new set of mini cell lengths in the RAM memory. The whole table can also ne transferred in a control message.

Instead of providing each cell with a fixed size length field which is used to indicate the mini cell size it is possible to use an implicit method of indicating the mini cell size which does not use any length field in the mini cell header. According to the implicit method of indicating mini cell sizes, information relating to the sizes is resident within the system network. Instead of using a dedicated field to indicate the cell size an existing field in the mini cell header is used. In the preferred embodiment of the invention mini cell sizes are mapped on the identities of established connections. Accordingly sizes are not global but connection oriented.

The identity of a connection is given by the CID field of a connection. In Figure 20 the mini cell header 7 is shown to comprise a CID field 71. The actual size of the CID field 71 depends on the system but generally two octets should be sufficient. By using the same mapping method as described in connection with Figures 6 and 7 a mapping table 72 results.

Accordingly the fixed length field 11 has been discarded. This will increase the band width efficiency. The CID value is

used as address to the RAM memory 70 in Figure 17 and is provided by the control system 80. So, instead of latching the length field 11 in the register 30, the CID value is latched in register 30 and is used as address to the RAM memory 70. In this manner there will be a relation between the identity of the established connection and the length of the mini cells used in the connection. Accordingly no additional memory places are needed for storing the relation between a CID and a size of the mini cell associated with said CID.

At set up of a connection the control system 80 will receive a message which requests (a) that a connection should be set up between to identified end points and (b) that this connection shall use mini cells having a size of X octets. X is supposed to be an integer selected among the available cell sizes. Next the control circuit selects a free CID among logical addresses provided by the ATM network. For the sake of the example CID=7 is selected. The control system 80 will now use 7 as an address to the RAM memory 70 and will write at this address the mini cell size X. The cell header reading device shown in Figure 17 will then operate in the same manner as described. It should be noted that the mapping takes place at connection set-up.

It should be noted that one and the same CID may relate to several different mini cell sizes depending on the fact that cells having the same CID can be transported on different virtual connections VC:s. This is illustrated in Figure 22 wherein a typical address structure used in an ATM network is shown. To each physical link, referred to as physical route, in the ATM network, there is a physical link table 140 having a number of entries, for example the indicated entries 0-23. To each physical link is associated a respective VPI/VCI (virtual path/virtual identifier) table 150. As an example